U.S. Application Serial No.   11
Docket No. YOR920010067US1
(YOR.271)

REMARKS

Claims 1-40 are all the claims presently pending in the application.

Claims 24, 26, and 34 have been amended merely to make minor editorial changes.

It is noted that the claim amendments are made only for more particularly pointing out

the invention, and <u>not</u> for distinguishing the invention over the prior art, narrowing the claims or

for any statutory requirements of patentability. Further, Applicants specifically state that no

amendment to any claim herein should be construed as a disclaimer of any interest in or right to

an equivalent of any element or feature of the amended claim.

Claims 1-26 and 28-40 stand rejected under 35 U.S.C. § 103(a) as being obvious from

Fulton, III et al. (U.S. Patent No. 5,715,386; hereinafter "Fulton") in view of Garg ("A

Methodology for Detection and Estimation of Software Aging", published November 1998.

Claim 27 stands rejected under 35 U.S.C. § 103(a) as being obvious over Fulton in view of Garg,

and further in view of Murphy, et al. (U.S. Publication No. 2003-015084 A1, filed January 10,

2000).

These rejections are respectfully traversed in the following discussion.

I.   THE CLAIMED INVENTION

In an illustrative, non-limiting aspect of the present application, as defined by

independent claim 1, a method of <u>reducing a time for a computer system to recover from a</u>

<u>degradation of performance in a hardware or a software</u> in at least one first node of the computer

system, includes <u>monitoring a state</u> of the at least one first node, and <u>based on the monitoring,</u>

<u>transferring a state</u> of the at least one first node to a second node <u>prior to the degradation in</u>

<u>performance of the hardware or the software</u> of the at least one first node.

U.S. Application Serial No.   12
Docket No. YOR920010067US1
(YOR.271)

In another exemplary aspect of the invention, as defined by independent claim 19, a

method of <u>reducing a lack of performance in a computer system</u> having at least one primary node

and a secondary node, includes <u>determining whether a failure or lack of performance is</u>

<u>imminent,</u> based on said determining, commanding a secondary node to <u>start an application</u> if it

is not already running, <u>and to begin reading a state and redo log</u> from a memory coupled to said

primary node and said secondary node, commanding the secondary node to <u>apply the redo log to</u>

<u>its state,</u> commanding the primary node to <u>begin mirroring its dynamic state updates</u> to the

secondary node as they occur, <u>such that the secondary node's state is brought completely up to</u>

<u>date with said primary node,</u> judging whether the primary node has failed, and based on said

judging, making the secondary node become the primary node.

Independent claims 26-28 and 34-40 recite somewhat similar features as independent

claims 1 and 19.

In conventional devices, when a computer system suffers an unplanned failure, <u>a certain</u>

<u>amount of time is required to recover from the failure. This outage duration is undesirable.</u>

For example, if the computer is a single node, stand-alone computer system, it must

reboot and restart its application. If the computer is part of a multi-node high availability cluster

architecture, it must failover (i.e., transfer) the application to another node in the cluster. During

this recovery time, after either rebooting or failing-over the application to another node in a

cluster environment, the recovering system must reload a stale copy of its state from disk, load a

transaction redo log from disk, and attempt to reconstruct an up-to-date copy of that state by

replaying that transaction redo log against the stale state (e.g., see specification at page 2, lines 2-

11).

The claimed invention, on the other hand, provides an exemplary method (and system) for proactively reducing the outage duration by using the predicted outages to proactively trigger and manage existing failure recovery functionality (e.g., see specification at page 1, lines 13-16).

Particularly, the claimed invention exploits the ability to predict software outages or hardware failures to proactively migrate the state needed to quickly recover from an imminent outage onto another computer system (e.g., such as another node in a cluster environment) or a persistent storage medium (e.g., such as a hard disk) (e.g., see specification at page 5, lines 10-18). According to the present invention, in an exemplary, non-limiting aspect, the system that was about to fail (e.g., the "failing computer") could "prime" another computer (e.g., the "failover target") by directing it to read from disk the stale state and the redo log, and then direct the failover target computer to begin applying the redo log to the stale state. Moreover, upon discovery of its imminent demise, the failing computer could mirror all transactions to the failover target, bringing the failover target's state even more up-to-date. When the primary computer does finally fail, the failover target would have an up-to-date copy of the state, the lengthy reloading of state and redo log from disk would be avoided, the outage would be shortened, and the system availability would be improved (e.g., see specification at page 6, lines 7-12).

Thus, the present invention reduces outage duration by using the ability to predict outages to proactively trigger and execute functionality whose effect is to reduce that outage's duration. Further, in the case of a single node, the amount of time required to reconstruct the node's state after the outage has occurred can be reduced significantly (e.g., see specification at page 6, lines 7-12).

U.S. Application Serial No.   14
Docket No. YOR920010067US1
(YOR.271)

II.   THE PRIOR ART REJECTIONS

A.   Claims 1-26 and 28-40 stand rejected under 35 U.S.C. § 103(a) as being obvious

from Fulton in view of Garg.

The Examiner alleges that Fulton discloses all of the features of the claimed invention,

except for monitoring the state of at least one first node being used in the initiation of the state

transfer (e.g., see Office Action at page 2, lines 21-22). However, the Examiner alleges that

Garg makes up for the deficiencies of Fulton by allegedly disclosing the monitoring of a state of

an important resource and using this to determine a measurement to indicate a time for software

rejuvenation (e.g., see Office Action at page 2, lines 23-24).

The Examiner alleges that it would have been obvious to use the monitoring of Garg in

the invention of Fulton because Garg indicates that the "Estimated time to exhaustion" metric

would have been valuable to indicate measurements that indicate a time when a system is

approaching degraded mode operation.

For the following reasons, Applicants respectfully submit that it would not have been

obvious to combine Fulton and Garg to arrive at the claimed invention. Moreover, even

assuming *arguendo* that such a combination would have been obvious, Applicants respectfully

submit that the resulting combination of Fulton and Garg would not disclose or suggest all of the

novel and unobvious features of the claimed invention. Thus, Applicants respectfully traverse

this rejection.

The claimed invention relates to an exemplary method (and system) for proactively

reducing the outage duration by using the predicted outages to proactively trigger and manage

existing failure recovery functionality (e.g., see specification at page 1, lines 13-16).

U.S. Application Serial No.   15
Docket No. YOR920010067US1
(YOR.271)

For example, independent claim 1 recites, *inter alia*, a method of reducing a time for a

computer system to recover from a degradation of performance in a hardware or a software in at

least one first node of said computer system, including:

> monitoring a state of said at least one first node; and
> based on said monitoring, transferring a state of said at least
> one first node to a second node prior to said degradation in
> performance of said hardware or said software of said at least one
> first node (emphasis added).

On the other hand, independent claim 19 recites, *inter alia*, a method of reducing a lack

of performance in a computer system having at least one primary node and a secondary node,

including:

> determining whether a failure or lack of performance is
> imminent;
> based on said determining, commanding a secondary node to
> start an application if it is not already running, and to begin reading
> a state and redo log from a memory coupled to said primary node
> and said secondary node;
> commanding the secondary node to apply the redo log to its
> state; .
> commanding the primary node to begin mirroring its dynamic
> state updates to the secondary node as they occur, such that the
> secondary node's state is brought completely up to date with said
> primary node;
> judging whether the primary node has failed; and
> based on said judging, making the secondary node become the
> primary node (emphasis added).

Thus, according to the novel and unobvious features of the present invention, a system

that is about to fail (e.g., the first node, or the "failing computer", etc.) can "prime" another

computer (e.g., the secondary node, or the "failover target", etc.) by directing it to read from disk

the stale state and the redo log, and then direct the failover target computer to begin applying the

redo log to the stale state (e.g., see specification at page 5, lines 19-23).

Moreover, upon discovery of its imminent demise, the failing computer could mirror all transactions to the failover target, bringing the failover target's state even more up-to-date. When the primary computer does finally fail, the failover target would have an up-to-date copy of the state, the lengthy reloading of state and redo log from disk would be avoided, the outage would be shortened, and the system availability would be improved (e.g., see specification at page 6, lines 7-12).

Thus, the present invention reduces outage duration by using the ability to predict outages to proactively trigger and execute functionality whose effect is to reduce that outage's duration. Further, in the case of a single node, the amount of time required to reconstruct the node's state after the outage has occurred can be reduced significantly (e.g., see specification at page 6, lines 7-12).

Applicants submit, however, that neither Fulton nor Garg discloses, suggests, or for that matter, even mentions reducing outage duration by using the ability to predict outages to proactively trigger and execute functionality whose effect is to reduce that outage's duration, as in the claimed invention.

Instead, Fulton relates to software rejuvenation techniques which improve reliability of a software system by periodically stopping execution of the system and restarting the execution with a "clean" state (e.g., see Fulton at Abstract; see also column 2, lines 11-17; emphasis added). That is, Fulton merely discloses the concept of rejuvenating the software.

However, contrary to the Examiner's position, Fulton does not disclose or suggest "*a method for reducing a time for a computer system to recover from a degradation of performance in a hardware or a software*" (e.g., see Office Action at page 2, lines 14-17; *citing* Fulton at column 2, lines 11-23).

U.S. Application Serial No.   17
Docket No. YOR920010067US1
(YOR.271)

Indeed, at column 2, lines 11-23, Fulton does <u>not</u> disclose, suggest, or even mention

<u>reducing outage duration</u> or <u>reducing a time for a computer system to recover</u> from a degradation

of performance, as in the claimed invention. Instead, Fulton simply states that:

> The invention is based on the observation that the probability
> that an application will fail <u>increases with the length of time that</u>
> <u>application continues to run</u>. Consequently, <u>failure can be</u>
> <u>prevented by periodically stopping execution of the application and</u>
> <u>restating the execution with clean internal state</u>. That procedure is
> termed herein <u>rejuvenation</u>. One way of rejuvenating an
> application is to take advantage of the fact that a newly-created
> process has clean internal state. Thus, <u>an application can be</u>
> <u>rejuvenated by terminating the process</u> which is presently
> executing the application <u>and restarting the application</u> so that it is
> executed by a new process.

(see Fulton at column 2, lines 11-23; emphasis added).

Accordingly, Fulton merely discloses the concept of rejuvenating the software, <u>not</u>

<u>reducing outage duration</u> or <u>reducing a time for a computer system to recover</u> from a degradation

of performance, as in the claimed invention (nor has the Examiner cited such disclosure in

Fulton).

In Fulton, an application is registered with the watchd daemon (i.e., the monitor), and

watchd monitors the application. If watchd detects that the process which is currently executing

the application is <u>dead or hung,</u> watchd <u>restarts the application</u> on a new process (e.g., see Fulton

at column 2, lines 33-35; emphasis added). That is, as the Examiner acknowledges, in Fulton the

watchd does <u>not</u> do anything until the application is <u>dead or hung</u>.

Moreover, Fulton does <u>not</u> disclose, suggest, or even mention <u>reducing the actual</u>

<u>duration of performing the rejuvenation</u> (e.g., the outage duration).

On the other hand, Garg relates to a method of <u>detecting and estimating aging in</u>

<u>operational software</u>. Garg monitors operating system resource usage and system activity and

U.S. Application Serial No.   18
Docket No. YOR920010067US1
(YOR.271)

discloses using an "Estimated time to exhaustion" metric <u>to compare the effect of aging on</u>

<u>different system resources</u> and also in the identification of important resources to monitor and

manage.

In other words, Garg is concerned with identifying the time (i.e., the <u>occasion</u>) for

performing the rejuvenation. That is, Garg is trying to predict <u>when</u> to perform the rejuvenation

such that the operational software <u>can be stopped and then restarted</u> in a clean state, hopefully

before the operational software actually fails (e.g., see Garg at page 1, column 2, first paragraph,

lines 7-10).

Similar to Fulton above, Garg also does <u>not</u>, however, disclose, suggest, or even mention

<u>reducing the actual duration of performing the rejuvenation</u> (e.g., the outage duration), but rather,

tries to predict <u>when</u> to perform the rejuvenation process. This has <u>no affect on the duration of</u>

<u>the rejuvenation process</u> (e.g., the outage duration), and Garg clearly does <u>not</u> disclose, suggest,

or even mention that such is the case.

Instead, Garg specifically states that the disclosed approach is "*a first step towards*

<u>predicting aging related failure occurrences</u>, and may help us in developing a strategy for

software fault-tolerance approaches, such as software rejuvenation, triggered by actual

measurements" (e.g., see page 10, first paragraph; emphasis added).

For the foregoing reasons, Applicants submit that the methods of both Fulton and Garg

are entirely unrelated to <u>limiting the duration of the rejuvenation process</u>, or for that matter,

<u>limiting the duration for failure recovery processes</u>, as in the claimed invention.

Accordingly, Applicants respectfully submit that it would <u>not</u> have been obvious to

combine Fulton and Garg <u>to arrive at the claimed invention</u> because neither Fulton nor Garg

even contemplates the problems addressed and solved by the claimed invention.

U.S. Application Serial No.   19
Docket No. YOR920010067US1
(YOR.271)

Moreover, Applicants submit that, even assuming *arguendo* that it would have been obvious to modify Fulton in view of Garg, the resulting combination clearly would <u>not</u> disclose or suggest the claimed invention, which <u>reduces outage duration</u> by using the ability to predict outages to proactively trigger and execute functionality whose effect is to reduce that outage's duration, as in the claimed invention.

Instead, the combination would result (at best) in <u>a rejuvenation process</u> as described in Fulton being performed <u>on the occasion</u> described by Garg, <u>not</u> a method that reduces the actual time for the computer system to recover (e.g., the outage duration) from a degradation of performance (or a rejuvenation process), as in the claimed invention.

For the foregoing reasons, Applicants respectfully submit that Fulton and Garg, either alone or in combination, do <u>not</u> disclose or suggest all of the novel and unobvious features of independent claims 1 and 19.

Applicants submit that independent claims 26-28 and 34-40 recite somewhat similar features as independent claims 1 and 19, and therefore, are patentable over Fulton and Garg for somewhat similar reasons as those set forth above.

Moreover, dependent claims 2-18, 20-25, and 29-33 also are patentable by virtue of their respective dependencies from independent claims 1, 19, and 28, as well as for the additional features recited therein.

Accordingly, the Examiner respectfully is requested to withdraw the rejection of claims 1-40 and permit these claims to pass to immediate allowance.

B.   Claim 27 stands rejected under 35 U.S.C. § 103(a) as being obvious over Fulton in view of Garg, and further in view of Murphy.

U.S. Application Serial No.   20
Docket No. YOR920010067US1
(YOR.271)

The Examiner alleges that it would have been obvious t combine Fulton, Garg, and

Murphy to arrive at the claimed invention. However Applicants respectfully disagree with the

Examiner's position, and therefore, traverse this rejection.

For example, independent claim 27 recites, *inter alia*, a method of <u>reducing a degradation</u>

<u>period</u> of a Web hosting machine, including:

> <u>monitoring a performance</u> of said Web hosting machine; and
> <u>transferring a state</u> of said Web hosting machine to a second
> machine <u>when a degradation of said performance occurs</u> in said
> Web hosting machine (emphasis added).

For somewhat similar reasons as those set forth above, Applicants submit that it clearly

would <u>not</u> have been obvious to combine Fulton and Garg <u>to arrive at the claimed invention,</u>

since neither Fulton nor Garg discloses, suggests, or even mentions "reducing a degradation

period" as claimed.

Moreover, Murphy would <u>not</u> have made up for the deficiencies of Fulton and Garg.

Indeed, Murphy is not even relied upon for the features of <u>reducing a degradation period</u> or

<u>reducing a time to recover from a degradation of performance</u> in a hardware or a software, as

claimed. Instead, Murphy is relied upon for showing "*a node for a Web hosting machine*" (see

Office Action at page 23, lines 7-8).

Thus, Applicants submit that Fulton, Garg, and Murphy, either alone or in combination,

do <u>not</u> disclose or suggest all of the novel and unobvious features of the claimed invention.

Therefore, the Examiner is requested to withdraw this rejection and permit claim 27 to

pass to immediate allowance.

U.S. Application Serial No.   21
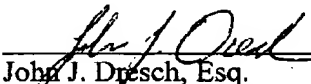Docket No. YOR920010067US1
(YOR.271)

III.   CONCLUSION

In view of the foregoing, Applicants submit that claims 1-40, all the claims presently

pending in the application, are patentably distinct over the prior art of record and are in condition

for _allowance_. The Examiner is respectfully requested to pass the above application to issue at

the earliest possible time.

Should the Examiner find the application to be other than in condition for allowance, the

Examiner is requested to contact the undersigned at the local telephone number listed below to

discuss any other changes deemed necessary in a telephonic or personal interview.

The Commissioner is hereby authorized to charge any deficiency in fees or to credit any

overpayment in fees to Assignee's Deposit Account No. 50-0510.

                                         Respectfully Submitted,

Date: _September 30, 2004_

                                         John J. Dresch, Esq.
                                         Registration No. 46,672

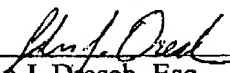   Sean M. McGinn, Esq.
                                         Registration No. 34,386

McGinn & Gibb, PLLC
8321 Old Courthouse Road, Suite 200
Vienna, VA 22182-3817
(703) 761-4100
Customer No. 21254

U.S. Application Serial No.   22
Docket No. YOR920010067US1
(YOR.271)

CERTIFICATE OF TRANSMISSION

I certify that I transmitted via facsimile to (703) 872-9306 the enclosed Amendment

under 37 C.F.R. § 1.111 to Examiner Joshua A. Lohn on September 30, 2004.

John J. Dresch, Esq.
Registration No. 46,672
Sean M. McGinn, Esq.
Registration No. 34,386